

Exhibit K



Next: [Compression Scheme](#) Up: [Title Page](#) Previous: [Introduction](#) Index: [Full Text Index](#) Contents: [Conference Page](#)

Wavelet-based 3D Compression

CT Compression

Wavelets are mathematical tools for representing functions hierarchically and have recently had great impacts on several areas of computer graphics ([1], [2], [3], [6], [19], [21]). Among the several bases of wavelets, the Haar basis is the simplest and computationally cheapest because it can be implemented by a few additions, subtractions and shift operations. Hence it is very effective in applications that require fast decomposition and reconstruction. However, it does not perform as well in terms of quality as other popular wavelet bases such as Daubechies.

In our framework, we first partition a given 3D volumetric dataset into subblocks, called *unit blocks*, whose size is $16 \times 16 \times 16$. For each unit block, a three-dimensional wavelet transform is repeatedly applied twice. We use a simple wavelet based on the Haar basis whose eight-band filter bank can be expressed as: (Figure 2) where $c_{ijk}^{i,j,k=0,1}$ on the right side are eight coefficients in each $2 \times 2 \times 2$ subregion of a unit block. c_{lll} represents their average, and the remaining coefficients on the left side are the detail values corresponding to the filter sequences, (for example, c_{hlh} is obtained by applying the high-pass filter, the low-pass filter, then the high-pass filter.) Two applications of the 3D wavelet transforms for each unit block are enough, considering that smaller number of transforms results in faster reconstruction, and that most of the data $63/64 (= 1 - (1/8^2))$ are already decomposed into detail coefficients. The decomposition process converts each unit block into a decomposed version that can be stored in an array with the same number of elements, using a proper ordering of the coefficients (Figure 2).

A typical wavelet compression algorithm has three basic components: transform, quantization and encoding. The wavelet coefficients are then quantized to restrict the values of the coefficients to a limited number of possibilities. Note that usually all of the information loss occur in this stage. Then the encoding stage takes the string of symbols coming from the quantizer, and attempts to represent the data stream as efficiently as possible without loss. Popular variable length coders, such as Huffman or arithmetic coders, work well. However, such techniques are not appropriate for the situation where an individual data item must be quickly reconstructed in an arbitrary sequence ([5], [18]).

Since the wavelet transforms are applied twice for decomposition, each $4 \times 4 \times 4$ subregion of a unit block can be considered as represented by an octree in Figure 5. The reconstruction process is the reverse of decomposition. For the Haar wavelets we use, the reconstruction formulae (Figure 3).

Shapiro implemented an encoding scheme, called *zerotree encoding*, which greatly improved the performance of the wavelet encoder, but was much slower ([20]). Variations of zerotree encoding scheme were proposed to show better performances ([14], [15], [16], [17]). In [9] and [10], we proposed effective encoding scheme for very large volume data compression, showed abilities of high compression ratio and fast random access by performance analysis.

RGB Compression

The RGB color images, stacked together, can be regarded as volume data whose voxels are vector-valued (r, g, b) . Replacing the coefficients in decomposition and reconstruction formulae by vectors, we can use the transform formulae and compression scheme of CT data without modification. Since we use 24-bit for (r, g, b) , all the wavelet coefficients are stored in a 3-byte stream. By quantizing vectors in the 3-byte stream to 1-byte indices of color table ([7]), we can improve compression ratio.

Note that we used L^2 norm for CT and RGB data compression.



Next: [Compression Scheme](#) Up: [Title Page](#) Previous: [Introduction](#) Index: [Full Text Index](#) Contents: [Conference Page](#)

